

**PATENT**  
**IBM Docket No. CH9-2001-0018**

**REMARKS**

**Status:**

Claim 1-3, 7-14, and 17-19 stand rejected under 35 U.S.C. §102(e) as being anticipated by the teaching of U. S. Publication No. 2003/0028686A1 to Schwabe. And, claims 4-6 and 15-16 stand rejected under 35 U.S.C. §103(a) as being unpatentable considering the teaching of the Schwabe reference in view of the teaching of U. S. Publication No. 2004/0068726A1 to Levy

Claims 1-19 are presented for reconsideration as is explained in the analysis that follows.

**Analysis:**

Looking first to the Schwabe teaching it appears that the focus is on converting neutral code such as JAVA code (class code 70 of Schwabe Fig. 4A) to CAP files 74 that are suited to resource constrained devices such as smart cards. To reduce storage requirements tokens 84 and export files 80 are created.

This is much like the step D of Applicants Figure creating the CAP file 20 and the export file 23. But the conversion of focus for Applicant's claimed invention is at step A (see Figure) converting the compressed CAP file back to a normal file semantically equivalent to the CAP file, such as a JAVA code file. Applicant has recognized that by so converting the compressed code that a semantic equivalent is obtained, it is possible to use normal verification tools to assure that the rules for the neutral code, say JAVA rules, are adhered to. Applicant in a preferred embodiment then signs the CAP file for both verifications, if successful, the original uncompressed code and the uncompressed CAP code created at step A.

Schwabe teaches a compressor converter for creating CAP files (compare element 15 of Applicants Figure) but Applicant's claims emphasize the conversion back to uncompressed code which involves steps A1 and A2. They further emphasize

**PATENT**  
**IBM Docket No. CH9-2001-0018**

verification step B performed on the expanded code. This is code purposely expanded to uncompressed form to allow a test of the code after compression using standard tools to be sure rules are not inadvertently violated in performing the compression step.

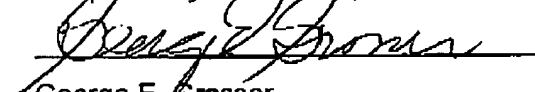
Where in the Schwabe teaching is there anything suggesting Applicants step A1 where compressed code and export file information are used to produce expanded code and step A2 where mapper 17 builds a semantically identical JAVA code file 40. Without the created file, it follows, there is no verification of such file. Both this conversion and the verification of the specially created file are emphasized in all of the claims (see, e.g., claim 1 portions a) and b) )

Without the file and verification, it follows that, there is no second cryptographic signature file created upon the successful verification of the file (see e.g, claims. 4-6).

The Levy teaching does not overcome these deficiencies. It teaches verification of bytecode. A significant challenge, but not what Applicant has claimed. This is the compressed CAP code that is being verified and authenticated. This is not a teaching of verification of specially expanded CAP files; file expanded to be, for example, JAVA code files which are testable with standard tools.

In accordance with the foregoing, it is believed that the claims clearly identify inventive content over the prior art. Consequently, withdrawal of the rejections of the claims is respectfully solicited along with notice that this case is in condition for allowance.

Respectfully Submitted,

  
George E. Grosser  
Reg. No. 25,629I

c/o IBM Corp.  
Dept. T81/Bldg. 503 PO Box 12195  
Research Triangle Park, NC 27709  
(919)968-7847 Fax 919-254-4330  
EMAIL: gegch@prodigy.net